

Введение в искусственный интеллект.

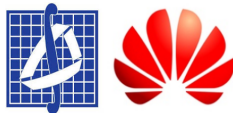
Машинное обучение

Лекция 1. Вводная

Бабин Д.Н., Иванов И.Е., Петюшко А.А.

кафедра Математической Теории Интеллектуальных Систем

18 февраля 2020 г.



① Организационные вопросы



- 1 Организационные вопросы
- 2 Историческая справка

- 1 Организационные вопросы
- 2 Историческая справка
- 3 Постановка основных задач машинного обучения



- 1 Организационные вопросы
- 2 Историческая справка
- 3 Постановка основных задач машинного обучения
- 4 Тестирование моделей, выбор лучшей



- 1 Организационные вопросы
- 2 Историческая справка
- 3 Постановка основных задач машинного обучения
- 4 Тестирование моделей, выбор лучшей
- 5 Декомпозиция ошибки, недообучение и переобучение





Руководитель курса: д.ф.-м.н. Бабин Дмитрий Николаевич



Лектор: к.ф.-м.н. Иванов Илья Евгеньевич



Лектор: к.ф.-м.н. Петюшко Александр Александрович



- Авторы имеют более 15 лет опыта участия в проектах, связанных с машинным обучением и компьютерным зрением
- Являются постоянными участниками группы распознавания образов кафедры MaTIC
- В качестве научных консультантов работают или работали с такими крупнейшими российскими и международными компаниями как Нейроком, LSI Research, Fotonation, Huawei и др.



- В данный момент времени авторы ведут исследования в области компьютерного зрения в московском научно-исследовательском центре Хуавэй
- Данный курс является частью программы **SHARE**
 - **SHARE** = School of Huawei Advanced Research Education
 - **SHARE** = Школа опережающего научного образования Хуавэй
 - e-mail: share@intsys.msu.ru
 - Канал SHARE: https://t.me/joinchat/AAAAAE_r4XKzEDaUKy1FwA



Зачем посещать этот курс

- 1 Специалисты по машинному обучению и анализу данных сейчас очень востребованы



Зачем посещать этот курс

- 1 Специалисты по машинному обучению и анализу данных сейчас очень востребованы
- 2 Шанс максимально использовать своё образование



Зачем посещать этот курс

- 1 Специалисты по машинному обучению и анализу данных сейчас очень востребованы
- 2 Шанс максимально использовать своё образование
- 3 Для лучших студентов возможны стажировки и бонусы от партнеров



Зачем посещать этот курс

- 1 Специалисты по машинному обучению и анализу данных сейчас очень востребованы
- 2 Шанс максимально использовать своё образование
- 3 Для лучших студентов возможны стажировки и бонусы от партнеров
- 4 И наконец, это просто интересно!



Что же такое искусственный интеллект?

Естественный интеллект (человек)

- Может воспринимать информацию, ее анализировать, принимать решения на основе анализа



Что же такое искусственный интеллект?

Естественный интеллект (человек)

- Может воспринимать информацию, ее анализировать, принимать решения на основе анализа

Искусственный интеллект

- (Сильный) то же самое, что и естественный, только на месте человека — компьютер



Что же такое искусственный интеллект?

Естественный интеллект (человек)

- Может воспринимать информацию, ее анализировать, принимать решения на основе анализа

Искусственный интеллект

- (Сильный) то же самое, что и естественный, только на месте человека — компьютер
- (Слабый) алгоритм, способный обучиться на основе массива входных данных, чтобы затем выполнять задачу вместо человека



Общая структура курса

«Введение в компьютерный интеллект»

① Машинное обучение

- Необходимые основы для всего курса
- Предыдущий курс лекций: [Весна 2019](#)



Общая структура курса

«Введение в компьютерный интеллект»

1 Машинное обучение

- Необходимые основы для всего курса
- Предыдущий курс лекций: [Весна 2019](#)

2 Компьютерное зрение

- Извлечение информации из визуальных образов (изображений и видео)
- Предыдущий курс лекций: [Осень 2019](#)



Общая структура курса

«Введение в компьютерный интеллект»

1 Машинное обучение

- Необходимые основы для всего курса
- Предыдущий курс лекций: [Весна 2019](#)

2 Компьютерное зрение

- Извлечение информации из визуальных образов (изображений и видео)
- Предыдущий курс лекций: [Осень 2019](#)

3 Обработка естественного языка

- Извлечение информации из речи и текста



Общая структура курса

«Введение в компьютерный интеллект»

1 Машинное обучение

- Необходимые основы для всего курса
- Предыдущий курс лекций: [Весна 2019](#)

2 Компьютерное зрение

- Извлечение информации из визуальных образов (изображений и видео)
- Предыдущий курс лекций: [Осень 2019](#)

3 Обработка естественного языка

- Извлечение информации из речи и текста

4 Обучение с подкреплением

- Интерактивное взаимодействие со средой



- Предсказание стоимости недвижимости



Применение машинного обучения в реальной жизни

- Предсказание стоимости недвижимости
- Предсказание платёжеспособности клиента



Применение машинного обучения в реальной жизни

- Предсказание стоимости недвижимости
- Предсказание платёжеспособности клиента
- Предсказание оттока клиентов



Применение машинного обучения в реальной жизни

- Предсказание стоимости недвижимости
- Предсказание платёжеспособности клиента
- Предсказание оттока клиентов
- Классификация заболеваний



Применение машинного обучения в реальной жизни

- Предсказание стоимости недвижимости
- Предсказание платёжеспособности клиента
- Предсказание оттока клиентов
- Классификация заболеваний
- Предсказание клика пользователя по рекламному баннеру



Применение машинного обучения в реальной жизни

- Предсказание стоимости недвижимости
- Предсказание платёжеспособности клиента
- Предсказание оттока клиентов
- Классификация заболеваний
- Предсказание клика пользователя по рекламному баннеру
- И многие другие задачи...



Что будет в этом курсе

Теоретическая часть

- Постановка задач машинного обучения. Тестирование и метрики качества
 - Precision / Recall, TPR / FPR, ROC, AUC, Cross-Validation, ...
- Методы классификации и оптимизации
 - SVM, Random Forest, Decision Tree, Stochastic Gradient Descent, ...
- Методы восстановления регрессии
 - Linear Regression, Elastic Net, Ridge Regression, LASSO, ...
- Композиции алгоритмов
 - Bootstrapping, Bagging, Boosting, AdaBoost, GBoost, ...



Что будет в этом курсе

Теоретическая часть

- Постановка задач машинного обучения. Тестирование и метрики качества
 - Precision / Recall, TPR / FPR, ROC, AUC, Cross-Validation, ...
- Методы классификации и оптимизации
 - SVM, Random Forest, Decision Tree, Stochastic Gradient Descent, ...
- Методы восстановления регрессии
 - Linear Regression, Elastic Net, Ridge Regression, LASSO, ...
- Композиции алгоритмов
 - Bootstrapping, Bagging, Boosting, AdaBoost, GBoost, ...

Практическая часть

- Обработка и анализ данных на python
 - Scikit-Learn, Numpy, Pandas, ...
- Соревнования по машинному обучению

Чего не будет в этом курсе

- Глубокое обучение

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Чего не будет в этом курсе

- Глубокое обучение
- Частичное обучение

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Чего не будет в этом курсе

- Глубокое обучение
- Частичное обучение
- Методы ранжирования

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Чего не будет в этом курсе

- Глубокое обучение
- Частичное обучение
- Методы ранжирования
- Прогнозирование временных рядов

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Чего не будет в этом курсе

- Глубокое обучение
- Частичное обучение
- Методы ранжирования
- Прогнозирование временных рядов
- Рекомендательные системы

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Чего не будет в этом курсе

- Глубокое обучение
- Частичное обучение
- Методы ранжирования
- Прогнозирование временных рядов
- Рекомендательные системы
- Цифровая обработка сигналов¹ и изображений²

¹См. курс “Математические основы цифровой обработки сигналов”, Мазуренко И. Л.

²См. курс “Математические основы цифровой обработки изображений”, Мазуренко И. Л.

Оценки за курс

- Оценки за курс будут выставляться в соответствии с данными о посещении и набранными баллами за выполнение домашних заданий.



Оценки за курс

- Оценки за курс будут выставляться в соответствии с данными о посещении и набранными баллами за выполнение домашних заданий.
- В ходе курса будут предложены домашние задания трёх типов:
 - теоретические
 - практические
 - соревнования



Оценки за курс

- Оценки за курс будут выставляться в соответствии с данными о посещении и набранными баллами за выполнение домашних заданий.
- В ходе курса будут предложены домашние задания трёх типов:
 - теоретические
 - практические
 - соревнования
- В конце семестра состоится экзамен, на котором при желании можно будет повысить свою оценку



- Оценки за курс будут выставляться в соответствии с данными о посещении и набранными баллами за выполнение домашних заданий.
- В ходе курса будут предложены домашние задания трёх типов:
 - теоретические
 - практические
 - соревнования
- В конце семестра состоится экзамен, на котором при желании можно будет повысить свою оценку
- Предварительная шкала оценок:

Оценка	Процент выполненных заданий
Отлично	80 %
Хорошо	60 %
Зачет	40 %



- Оценки за курс будут выставляться в соответствии с данными о посещении и набранными баллами за выполнение домашних заданий.
- В ходе курса будут предложены домашние задания трёх типов:
 - теоретические
 - практические
 - соревнования
- В конце семестра состоится экзамен, на котором при желании можно будет повысить свою оценку
- Предварительная шкала оценок:

Оценка	Процент выполненных заданий
Отлично	80 %
Хорошо	60 %
Зачет	40 %

За посещение каждого занятия балл увеличивается на 1%.



Полезные ресурсы

- Страница курса: <https://github.com/mlcoursemm/mlcoursemm2020spring>
- Главный ресурс по курсам “Введение в компьютерный интеллект”:
<https://github.com/mlcoursemm>
- Телеграмм-канал: <https://t.me/joinchat/AAAAAEUmx5cJL0dLXs0t8g>
- Группа обсуждения: <https://t.me/joinchat/B2iAGkx8IrWpQdM3LL-H-w>
- Почта курса: mlcoursemm@gmail.com

“Введение в компьютерный интеллект”

Канал с новостями



Что такое машинное обучение

В 1959 году Артур Самуэль (Arthur Samuel) ввел в научный обиход термин “машинное обучение”.

Общее определение

Машинное обучение — процесс, в результате которого компьютеры способны показать поведение, которое в них не было явно запрограммировано.



Что такое машинное обучение

В 1959 году Артур Самуэль (Arthur Samuel) ввел в научный обиход термин “машинное обучение”.

Общее определение

Машинное обучение — процесс, в результате которого компьютеры способны показать поведение, которое в них не было явно запрограммировано.

В 1997 году Том М. Митчелл (Tom M. Mitchell) предложил формальное определение алгоритма машинного обучения.

Формальное определение

Считается, что **компьютерная программа обучается** по примерам E для некоторого набора задач T и метрики качества P , если качество ее работы на задачах из T , измеренное с помощью P , улучшается с использованием примеров E .



- Люди пытались предсказывать будущее на основе своего опыта с незапамятных времен
- Однако научную основу заложили теория вероятностей (в особенности статистика) и линейная алгебра (как инструмент).



- Люди пытались предсказывать будущее на основе своего опыта с незапамятных времен
- Однако научную основу заложили теория вероятностей (в особенности статистика) и линейная алгебра (как инструмент).
- **1795** Гаусс впервые применяет метод наименьших квадратов (МНК) для анализа астрономических наблюдений. В 1805 Лежандр впервые публикует этот метод для анализа формы Земли. В настоящее время МНК является простейшим способом решить переопределенную систему линейных уравнений.



- Люди пытались предсказывать будущее на основе своего опыта с незапамятных времен
- Однако научную основу заложили теория вероятностей (в особенности статистика) и линейная алгебра (как инструмент).
- **1795** Гаусс впервые применяет метод наименьших квадратов (МНК) для анализа астрономических наблюдений. В 1805 Лежандр впервые публикует этот метод для анализа формы Земли. В настоящее время МНК является простейшим способом решить переопределенную систему линейных уравнений.
- **1901** Карл Пирсон (Karl Pearson) изобрел метод главных компонент — главный метод уменьшения размерности данных.



- Люди пытались предсказывать будущее на основе своего опыта с незапамятных времен
- Однако научную основу заложили теория вероятностей (в особенности статистика) и линейная алгебра (как инструмент).
- **1795** Гаусс впервые применяет метод наименьших квадратов (МНК) для анализа астрономических наблюдений. В 1805 Лежандр впервые публикует этот метод для анализа формы Земли. В настоящее время МНК является простейшим способом решить переопределенную систему линейных уравнений.
- **1901** Карл Пирсон (Karl Pearson) изобрел метод главных компонент — главный метод уменьшения размерности данных.
- **1906** Андрей Андреевич Марков разрабатывает аппарат марковских цепей, который в 1913 применяет для исследования текста “Евгений Онегин”. Марковские цепи применяются для генерации и распознавания сигналов.



- **1950** Алан Тьюринг (Alan Turing) создает тест Тьюринга для оценки интеллекта компьютера.



- **1950** Алан Тьюринг (Alan Turing) создает тест Тьюринга для оценки интеллекта компьютера.
- **1951** Марвин Минский (Marvin Minsky) создал первую обучающуюся машину SNARC со случайно связанной нейросетью в 1951. В 1959 стал одним из сооснователей лаборатории искусственного интеллекта в MIT.



- **1950** Алан Тьюринг (Alan Turing) создает тест Тьюринга для оценки интеллекта компьютера.
- **1951** Марвин Минский (Marvin Minsky) создал первую обучающуюся машину SNARC со случайно связанной нейросетью в 1951. В 1959 стал одним из сооснователей лаборатории искусственного интеллекта в MIT.
- **1952** Артур Самуэль создает первую шашечную программу для IBM 701. В 1955 году Самуэль добавляет в программу способность к самообучению.



- **1950** Алан Тьюринг (Alan Turing) создает тест Тьюринга для оценки интеллекта компьютера.
- **1951** Марвин Минский (Marvin Minsky) создал первую обучающуюся машину SNARC со случайно связанной нейросетью в 1951. В 1959 стал одним из сооснователей лаборатории искусственного интеллекта в MIT.
- **1952** Артур Самуэль создает первую шашечную программу для IBM 701. В 1955 году Самуэль добавляет в программу способность к самообучению.
- **1958** Фрэнк Розенблатт (Frank Rosenblatt) придумал Персептрон — первую искусственную нейронную сеть и создал первый нейрокомпьютер “Марк-1”. New York Times: Персептрон — это “эмбрион электронного компьютера, который в будущем сможет ходить, говорить, видеть, писать, воспроизводить себя и осознавать свое существование”.



- **1950** Алан Тьюринг (Alan Turing) создает тест Тьюринга для оценки интеллекта компьютера.
- **1951** Марвин Минский (Marvin Minsky) создал первую обучающуюся машину SNARC со случайно связанной нейросетью в 1951. В 1959 стал одним из сооснователей лаборатории искусственного интеллекта в MIT.
- **1952** Артур Самуэль создает первую шашечную программу для IBM 701. В 1955 году Самуэль добавляет в программу способность к самообучению.
- **1958** Фрэнк Розенблатт (Frank Rosenblatt) придумал Персептрон — первую искусственную нейронную сеть и создал первый нейрокомпьютер “Марк-1”. New York Times: Персептрон — это “эмбрион электронного компьютера, который в будущем сможет ходить, говорить, видеть, писать, воспроизводить себя и осознавать свое существование”.
- **1963** Лоуренс Робертс (Lawrence Roberts) сформулировал тезисы компьютерного зрения в своей диссертации в MIT.



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.
- **1965** Опубликована одна из первых книг по машинному обучению (классификации образов) — Nilsson N. Learning Machines, McGraw Hill.



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.
- **1965** Опубликована одна из первых книг по машинному обучению (классификации образов) — Nilsson N. Learning Machines, McGraw Hill.
- **1966** Джозеф Вейценбаум (Joseph Weizenbaum) в 1966 году написал виртуального собеседника ELIZA, способного имитировать (а скорее, пародировать) диалог с психотерапевтом (своим названием программа обязана героине из пьесы Б. Шоу).



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.
- **1965** Опубликована одна из первых книг по машинному обучению (классификации образов) — Nilsson N. Learning Machines, McGraw Hill.
- **1966** Джозеф Вейценбаум (Joseph Weizenbaum) в 1966 году написал виртуального собеседника ELIZA, способного имитировать (а скорее, пародировать) диалог с психотерапевтом (своим названием программа обязана героине из пьесы Б. Шоу).
- **1967** Алексей Ивахненко и Валентин Лапа публикуют первый общий работающий обучающийся алгоритм для глубоких многослойных перцептронов для задач обучения с учителем.



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.
- **1965** Опубликована одна из первых книг по машинному обучению (классификации образов) — Nilsson N. Learning Machines, McGraw Hill.
- **1966** Джозеф Вейценбаум (Joseph Weizenbaum) в 1966 году написал виртуального собеседника ELIZA, способного имитировать (а скорее, пародировать) диалог с психотерапевтом (своим названием программа обязана героине из пьесы Б. Шоу).
- **1967** Алексей Ивахненко и Валентин Лапа публикуют первый общий работающий обучающийся алгоритм для глубоких многослойных перцептронов для задач обучения с учителем.
- **1986** Рина Дехтер (Rina Dechter) представила термин “Глубокое обучение” (Deep Learning) сообществу машинного обучения.



- **1963** Владимир Вапник и Алексей Червоненкис изобрели алгоритм SVM.
- **1965** Опубликована одна из первых книг по машинному обучению (классификации образов) — Nilsson N. Learning Machines, McGraw Hill.
- **1966** Джозеф Вейценбаум (Joseph Weizenbaum) в 1966 году написал виртуального собеседника ELIZA, способного имитировать (а скорее, пародировать) диалог с психотерапевтом (своим названием программа обязана героине из пьесы Б. Шоу).
- **1967** Алексей Ивахненко и Валентин Лапа публикуют первый общий работающий обучающийся алгоритм для глубоких многослойных перцептронов для задач обучения с учителем.
- **1986** Рина Дехтер (Rina Dechter) представила термин “Глубокое обучение” (Deep Learning) сообществу машинного обучения.
- **1997** Компьютер Deep Blue обыграл чемпиона мира по шахматам Гарри Каспарова.



- **2010** Основание компании DeepMind.



- **2010** Основание компании DeepMind.
- **2011** Эндрю Ын (Andrew Ng), Грег Коррадо (Greg Corrado) и Джефф Дин (Jeff Dean) основали Google Brain.



- **2010** Основание компании DeepMind.
- **2011** Эндрю Ын (Andrew Ng), Грег Коррадо (Greg Corrado) и Джефф Дин (Jeff Dean) основали Google Brain.
- **2011** Суперкомпьютер IBM Watson, оснащенный системой искусственного интеллекта, одержал победу в телевикторине Jeopardy! Его соперниками были маститые игроки Брэд Раттер (Bred Ratter) и Кен Дженнингс (Ken Jennings).



- **2010** Основание компании DeepMind.
- **2011** Эндрю Ын (Andrew Ng), Грег Коррадо (Greg Corrado) и Джефф Дин (Jeff Dean) основали Google Brain.
- **2011** Суперкомпьютер IBM Watson, оснащенный системой искусственного интеллекта, одержал победу в телевикторине Jeopardy! Его соперниками были маститые игроки Брэд Раттер (Bred Ratter) и Кен Дженнингс (Ken Jennings).
- **2014** В Facebook изобрели программный алгоритм DeepFace для распознавания лиц. Точность алгоритма составила 97%.



- **2010** Основание компании DeepMind.
- **2011** Эндрю Ын (Andrew Ng), Грег Коррадо (Greg Corrado) и Джефф Дин (Jeff Dean) основали Google Brain.
- **2011** Суперкомпьютер IBM Watson, оснащенный системой искусственного интеллекта, одержал победу в телевикторине Jeopardy! Его соперниками были маститые игроки Брэд Раттер (Bred Ratter) и Кен Дженнингс (Ken Jennings).
- **2014** В Facebook изобрели программный алгоритм DeepFace для распознавания лиц. Точность алгоритма составила 97%.
- **2016** Программа AlphaGo, разработанная гугловской компанией DeepMind, выиграла в четырех играх из пяти у чемпиона мира по игре в го корейца Ли Седоля (Lee Se-dol).



- **2010** Основание компании DeepMind.
- **2011** Эндрю Ын (Andrew Ng), Грег Коррадо (Greg Corrado) и Джефф Дин (Jeff Dean) основали Google Brain.
- **2011** Суперкомпьютер IBM Watson, оснащенный системой искусственного интеллекта, одержал победу в телевикторине Jeopardy! Его соперниками были маститые игроки Брэд Раттер (Bred Ratter) и Кен Дженнингс (Ken Jennings).
- **2014** В Facebook изобрели программный алгоритм DeepFace для распознавания лиц. Точность алгоритма составила 97%.
- **2016** Программа AlphaGo, разработанная гугловской компанией DeepMind, выиграла в четырех играх из пяти у чемпиона мира по игре в го корейца Ли Седоля (Lee Se-dol).
- **2016** При поддержке Илона Маска (Elon Musk) запущена некоммерческая исследовательская компания OpenAI.



Определения

- X — множество объектов
- Y — множество ответов
- $y : X \rightarrow Y$ — неизвестная зависимость



Способы машинного обучения

Определения

- X — множество объектов
- Y — множество ответов
- $y : X \rightarrow Y$ — неизвестная зависимость

Основные способы машинного обучения

- **С учителем**
 - Достаточное количество обучающего материала, то есть пар (x_i, y_i)
- Частичное обучение
 - Малое количество размеченных данных и много неразмеченных примеров x_i
- *Без учителя*
 - Нет размеченных пар, только примеры x_i
- С подкреплением
 - Формирование отклика на основе взаимодействия со средой

Постановка задачи обучения с учителем

- Дано:
 - $\{(x_1, y_1), \dots, (x_n, y_n)\} \subset X \times Y$ — обучающая выборка
- Найти
 - Решающую функцию $a : X \rightarrow Y$, которая приближает целевую зависимость y .



Постановка задачи обучения с учителем

- Дано:
 - $\{(x_1, y_1), \dots, (x_n, y_n)\} \subset X \times Y$ — обучающая выборка
- Найти
 - Решающую функцию $a : X \rightarrow Y$, которая приближает целевую зависимость y .
- Необходимо детализировать:
 - Как определяются объекты
 - Как задаются ответы
 - Что значит, что одна зависимость приближает другую



Как определяются объекты

Определение

Объект = совокупность признаков



Как определяются объекты

Определение

Объект = совокупность признаков

Типы признаков

- Бинарный признак
- Категориальный признак
- Порядковый признак
- Количественный признак



Задачи классификации

- Бинарная классификация $Y = \{-1, 1\}$ или $Y = \{0, 1\}$
- Многоклассовая классификация $Y = \{0, 1, \dots, M - 1\}$
- Многозначная бинарная классификация $Y = \{0, 1\}^M$



Задачи классификации

- Бинарная классификация $Y = \{-1, 1\}$ или $Y = \{0, 1\}$
- Многоклассовая классификация $Y = \{0, 1, \dots, M - 1\}$
- Многозначная бинарная классификация $Y = \{0, 1\}^M$

Задачи восстановления регрессии

$Y = \mathbb{R}$ или $Y = \mathbb{R}^n$



Определение

Функция потерь (loss function) $L(a, x)$ — величина ошибка алгоритма a на объекте x



Определение

Функция потерь (loss function) $L(a, x)$ — величина ошибка алгоритма a на объекте x

Функции потерь для задачи классификации

$L(a, x) = [a(x) \neq y]$ — индикатор ошибки



Определение

Функция потерь (loss function) $L(a, x)$ — величина ошибка алгоритма a на объекте x

Функции потерь для задачи классификации

$L(a, x) = [a(x) \neq y]$ — индикатор ошибки

Функции потерь для задач регрессии

$L(a, x) = (a(x) - y)^2$ — квадратичная ошибка

Как понять, что одна модель лучше другой?

Для этого используют независимое от **обучающего** множества множество, которое называется **ТЕСТОВЫМ**



Как понять, что одна модель лучше другой?

Для этого используют независимое от **обучающего** множества множество, которое называется **тестовым**

Зачем вообще это понимать?

- Существует множество алгоритмов машинного обучения и важно понимать, какой из них более применим в конкретной задаче
- Даже в рамках одной модели может существовать множество параметров



Как выбирать лучшую модель

Наивный подход

Обучить модели с различными параметрами и выбрать лучшую на тесте



Как выбирать лучшую модель

Наивный подход

Обучить модели с различными параметрами и выбрать лучшую на тесте

Минусы наивного подхода

- Так как тест обычно состоит из случайной подвыборки исходной выборки, то результат на тесте тоже является некоторым приближением случайной величины
- Если все модели тестировать на тестовом датасете и таким образом выбирать лучшую, то будет происходить неявное обучение на тесте, а на другом независимом тесте возможны сюрпризы



Как выбирать лучшую модель

Наивный подход

Обучить модели с различными параметрами и выбрать лучшую на тесте

Минусы наивного подхода

- Так как тест обычно состоит из случайной подвыборки исходной выборки, то результат на тесте тоже является некоторым приближением случайной величины
- Если все модели тестировать на тестовом датасете и таким образом выбирать лучшую, то будет происходить неявное обучение на тесте, а на другом независимом тесте возможны сюрпризы

Что же делать?

Чтобы неявно не обучиться на тестовых данных — надо использовать кросс-валидацию

Общая идея

Основная идея кросс-валидации состоит в разбиении обучающего множества на два непересекающихся множества (возможно многократном):

$$X^{learn} = X^{train} \sqcup X^{val}$$

На одном из них происходит обучение, а на другом происходит валидация модели.



Общая идея

Основная идея кросс-валидации состоит в разбиении обучающего множества на два непересекающихся множества (возможно многократном):

$$X^{learn} = X^{train} \sqcup X^{val}$$

На одном из них происходит обучение, а на другом происходит валидация модели.

Частные случаи

- 1 Простейшая кросс-валидация (Holdout) — однократное разделение множества



Train

Validation

Частные случаи

2 k-fold валидация³:

Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 1
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 2
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 3
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 4
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 5

Training data Test data

³Image source: <https://towardsdatascience.com>

Частные случаи

2 k-fold валидация³:

Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 1
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 2
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 3
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 4
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 5

Training data

Test data

3 Leave one out (LOO) валидация — частный случай k-fold валидации, если k равно мощности обучающего множества

³Image source: <https://towardsdatascience.com>

Частные случаи

2 k-fold валидация³:

Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 1
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 2
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 3
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 4
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Metric 5

Training data

Test data

- 3 Leave one out (LOO) валидация — частный случай k-fold валидации, если k равно мощности обучающего множества
- 4 Многократная k-fold валидация — повторение k-fold валидации несколько раз с разными разбиениями.

³Image source: <https://towardsdatascience.com>

Определение

Переобучение (overfitting) — нежелательное явление, возникающее при решении задач обучения по прецедентам, когда вероятность ошибки обученного алгоритма на объектах тестовой выборки оказывается существенно выше, чем средняя ошибка на обучающей выборке.

Переобучение возникает при использовании избыточно сложной модели



Определение

Переобучение (overfitting) — нежелательное явление, возникающее при решении задач обучения по прецедентам, когда вероятность ошибки обученного алгоритма на объектах тестовой выборки оказывается существенно выше, чем средняя ошибка на обучающей выборке.

Переобучение возникает при использовании избыточно сложной модели

Одна из основных причин возникновения

Избыточная сложность пространства параметров модели, “лишние” степени свободы используются для точной настройки на обучающую выборку



Переобучение

Определение

Переобучение (overfitting) — нежелательное явление, возникающее при решении задач обучения по прецедентам, когда вероятность ошибки обученного алгоритма на объектах тестовой выборки оказывается существенно выше, чем средняя ошибка на обучающей выборке.

Переобучение возникает при использовании избыточно сложной модели

Одна из основных причин возникновения

Избыточная сложность пространства параметров модели, “лишние” степени свободы используются для точной настройки на обучающую выборку

Один из основных методов обнаружения

Использование кросс-валидации

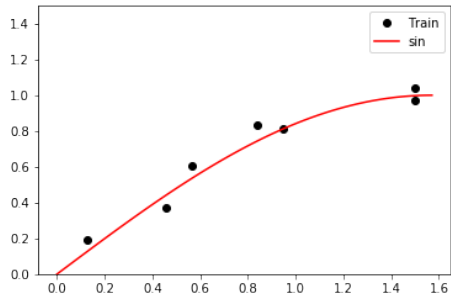
Определение

Недообучение (underfitting) – нежелательное явление, возникающее при решении задач обучения по прецедентам, когда алгоритм обучения не обеспечивает достаточно малой величины средней ошибки на обучающей выборке.

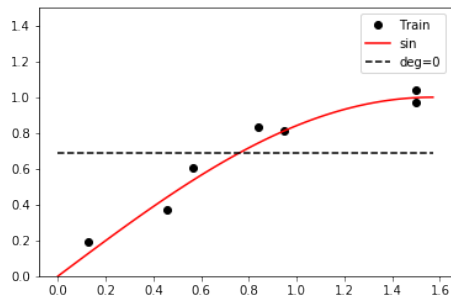
Недообучение возникает при использовании недостаточно сложных моделей



Примеры недообучения и переобучения



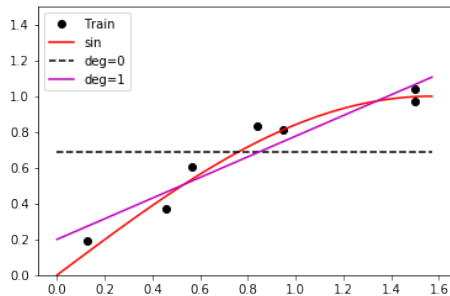
Примеры недообучения и переобучения



- Полином нулевой степени не может хорошо приближать зависимость в силу ограниченности параметров модели



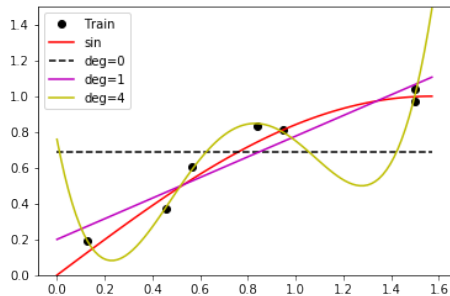
Примеры недообучения и переобучения



- Полином нулевой степени не может хорошо приближать зависимость в силу ограниченности параметров модели
- Линейная и квадратичная модели адекватно описывают закономерность



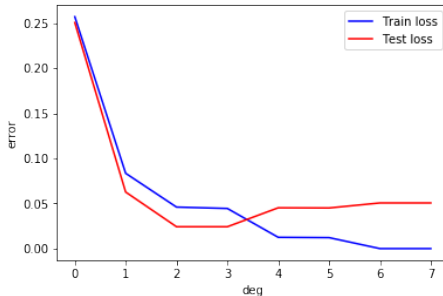
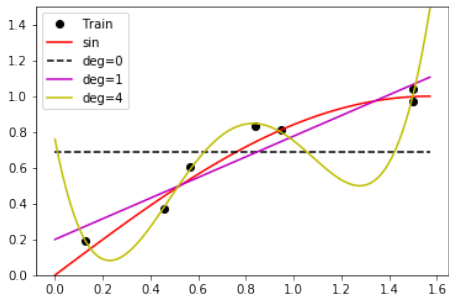
Примеры недообучения и переобучения



- Полином нулевой степени не может хорошо приближать зависимость в силу ограниченности параметров модели
- Линейная и квадратичная модели адекватно описывают закономерность
- Полиномы высоких степеней могут в точности пройти через точки обучающей выборки



Примеры недообучения и переобучения



- Полином нулевой степени не может хорошо приближать зависимость в силу ограниченности параметров модели
- Линейная и квадратичная модели адекватно описывают закономерность
- Полиномы высоких степеней могут в точности пройти через точки обучающей выборки



Определения

Пусть $y = y(x) = f(x) + \varepsilon$ — целевая зависимость, где $f(x)$ — детерминированная функция, $\varepsilon \sim N(0, \sigma^2)$ и $a(x)$ — алгоритм машинного обучения.



Определения

Пусть $y = y(x) = f(x) + \varepsilon$ — целевая зависимость, где $f(x)$ — детерминированная функция, $\varepsilon \sim N(0, \sigma^2)$ и $a(x)$ — алгоритм машинного обучения.

Полагаем, что ε и a — независимые. $Ey = Ef$, $Dy = D\varepsilon = \sigma^2$.

Разложение квадрата ошибки

$$\begin{aligned} E(y - a)^2 &= E(y^2 + a^2 - 2ya) = Ey^2 + Ea^2 - 2Eya = \\ &= Ey^2 + Ea^2 - 2E(f + \varepsilon)a = Ey^2 + Ea^2 - 2Efa - 2E\varepsilon a = \\ &= Ey^2 - (Ey)^2 + (Ey)^2 + Ea^2 - (Ea)^2 + (Ea)^2 - 2fEa = \\ &= Dy + Da + (Ey)^2 + (Ea)^2 - 2fEa = \\ &= Dy + Da + (E(f - a))^2 = \sigma^2 + \text{variance}(a) + \text{bias}^2(f, a) \end{aligned}$$

Определение

Разброс (variance) — дисперсия ответов алгоритмов $a(x)$.

Характеризует разнообразие алгоритмов (из-за случайности обучающей выборки, шума, стохастичности обучения и т.д.)



Определение

Разброс (variance) — дисперсия ответов алгоритмов $a(x)$.

Характеризует разнообразие алгоритмов (из-за случайности обучающей выборки, шума, стохастичности обучения и т.д.)

Определение

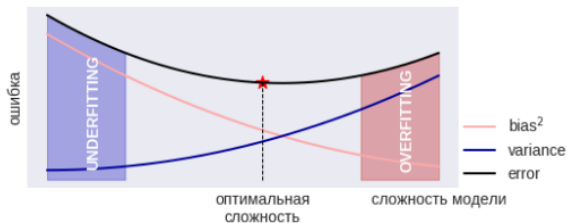
Смещение (bias) — матожидание разности между истинным ответом и выбранным алгоритмом.

Характеризует способность модели настраиваться на целевую зависимость



Модель оптимальной сложности

- Для простых моделей характерно недообучение
- Для сложных моделей характерно переобучение
- Оптимальная сложность модели где-то между⁴



⁴Image source: <https://dyakonov.org>

Спасибо за внимание!

